AMENDMENTS TO THE CLAIMS

Please Amend the claims according to the following listing. This listing of claims will replace all prior versions, and listings, of claims in the application:

LISTING OF CLAIMS

1. (Cancelled)

2. (Previously Presented) The system of claim 17, wherein the speculative data load is loaded in the pipeline.

3. (Previously Presented) The system of claim 17, wherein one or more of the data loads in the pipeline are not dependent on any specific data load and not selectively flagged.

4. (Previously Presented) The system of claim 17, wherein the flag is a bit within the instruction.

5. (Previously Presented) The system of claim 17, wherein the flag is attached to the instruction.

6. (Previously Presented) The system of claim 17, wherein each flagged instruction is flushed from the pipeline upon the determination of a misprediction for a data load.

7. (Previously Presented) The system of claim 17, wherein the fast-load data cache includes a directory.

8. (Previously Presented) The system of claim 17, wherein the fast-load data cache does not include a directory.

9. (Cancelled)

10. (Previously Presented) The method of claim 18, further comprising the step of loading the speculative data load into the pipeline.

11. (Previously Presented) The method of claim 18, wherein the step of selectively flagging the one or more instructions does not flag an instruction that is not dependent on a specific instruction.

12. (Previously Presented) The method of claim 18, wherein the step of selectively flagging a dependant instruction occurs through altering a bit within the dependant instruction.

13. (Previously Presented) The method of claim 18, wherein the step of selectively flagging the dependant instruction occurs through attaching a flag to the dependant instruction.

14. (Previously Presented) The method of claim 18, further comprising the step of flushing the flagged dependent instruction from the pipeline upon the determination of a misprediction of a corresponding data load.

15-16. (Cancelled)

17. (Currently Amended) A computer architecture, comprising:
at least one pipeline able to selectively load, execute and flush as series of instructions and capable of selectively flagging each of the series of instructions with a flag to

indicate dependence upon the load of a speculative instruction;

at least one fast-load data cache that loads at least one speculative data load relative to the speculative instruction;

a circuit that determines if the speculative data load is a misprediction; and

the pipeline being constructed so that if the speculative data load is a misprediction, then execution of ~~the~~ a dependant instruction is inhibited while the dependant instruction is in the pipeline, otherwise ~~executing~~ execution of the dependant instruction is not inhibited.

18. (Original) A method for executing instructions in a computer architecture that includes a pipeline, the method comprising the steps of:

loading a plurality of instructions into the pipeline;

selectively flagging a dependant instruction in the pipeline, thereby indicating that the dependant instruction depends upon the load of a speculative instruction

loading a speculative data load relative to the speculative instruction;

determining if the speculative data load is a misprediction; and

inhibiting execution of the dependant instruction while the dependant instruction is in the pipeline if the speculative data load is a misprediction, otherwise executing the dependant instruction.